

PBS Scheduling Policy

Category: Running Jobs with PBS

DRAFT

This article is being reviewed for completeness and technical accuracy.

This article gives a simplified explanation of the PBS scheduling policy on Pleiades and Columbia

PBS scheduling policies change frequently, in response to varying demands and workloads. The current policy (March 1, 2011), simplified, states that jobs are sorted in the following order: current mission directorate CPU use, job priority, queue priority, and job size (wide jobs first).

In each scheduling cycle, PBS examines the jobs in sorted order, starting a job if it can. If the job cannot be started immediately, it is either scheduled around or simply bypassed for this cycle.

There are numerous reasons why jobs won't start, such as:

- The queue is at its running job limit
- You are at your running job limit
- The queue is at its CPU limit
- The mission directorate is at its CPU share limit and the job cannot borrow from another mission
- Not enough CPUs are available

Notice that a high-priority job might be blocked by some limit, while a lower priority job, from a different user or asking for fewer resources, might not be blocked.

If your job is waiting in the queue, use the following commands to get some information about why it has not started running.

```
pfe1% qstat -s jobid  
or  
pfe1% qstat -f jobid | grep -i comment
```

On Pleiades, output from the following command shows the amount of resources (broken down into Harpertown, Nehalem, and Westmere processors) used and borrowed by each mission directorate, and the resources each mission is waiting for:

```
pfe1% /u/scicon/tools/bin/qs
```

The following command provides the order of jobs that PBS schedules to start at the current scheduling cycle. It also provides information regarding processor type(s), mission, and job priority:

```
pfe1% qstat -W o==+model,mission,pri -i
```

The policy described above could result in a large, high-priority job being blocked forever by a steady stream of smaller, low-priority jobs. To prevent jobs from languishing in the queues for an indefinite time, PBS reserves resources for the top N jobs (currently, N is 4), and doesn't allow lower priority jobs start if they would delay the start time of one of the top job ("backfilling"). Additional details are given below.

PBS Sorting Order

- **Mission shares**

Each NASA mission directorate is allocated a certain percentage of the CPUs in the system. (See [Mission Shares Policy on Pleiades](#) .) A job cannot start if that action would cause the mission to exceed its share, unless another mission is using less than its share and has no jobs waiting. In this case, the high-use mission can "borrow" CPUs from the lower-use mission for up to a specified time (currently, max_borrow is 4 hours).

So , if the job itself needs less than max_borrow hours to run, or if a sufficient number of other jobs from the high-use mission will finish within max_borrow hours to get back under its mission share, then the job can borrow CPUs.

When jobs are sorted, jobs from missions using less of their share are picked before jobs from missions using more of their share.

- **Job priority**

Job priority has three components. First is the native priority (the -p parameter to qsub or qalter). Added to that is the queue priority. If the native priority is 0, then a further adjustment is made based on how long the job has been waiting for resources. Waiting jobs get a "boost" of up to 20 priority points, depending on how long they have been waiting and which queue they are in.

This treatment is modified for queues assigned to the Exploration Systems Mission Directorate (ESMD). For those queues, job priority is set by a separate set of policies controlled by ESMD management.

- **Queue priority**

Some queues are given higher or lower priorities than the default. (Run "qstat -Q" to get current values.) Note that because the mission share is the most significant sort

criterion, job and queue priorities have little effect mission-to-mission.

- **Job size**

Jobs asking for more CPUs are favored over jobs asking for fewer. The reasoning is that, while it is easier for narrow jobs to fill in gaps in the schedule, wide jobs need help collecting enough CPUs to start.

Backfilling

As mentioned above, when PBS cannot start a job immediately, if it is one of the first N such jobs, PBS sets aside resources for the job before examining other jobs. That is, PBS looks at the currently running jobs to see when they will finish (using the wall-time estimates). From those finish times, PBS decides when enough resources (such as CPUs, memory, mission share, and job limits) will become available to run the top job.

PBS then creates a virtual reservation for those resources at that time. Now, when PBS looks at other jobs to see if they can start immediately, it also checks whether starting the job would collide with one of these reservations. Only if there are no collisions will PBS start the lower priority jobs.

This description applies to both Pleiades and Columbia, although the specific queues, priorities, mission percentages, and other elements differ between the two systems.

Article ID: 179

Last updated: 29 Jul, 2011

Computing at NAS -> Running Jobs with PBS -> PBS Scheduling Policy

<http://www.nas.nasa.gov/hecc/support/kb/entry/179?ajax=1>